

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 PROVA SCRITTA DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2012/2013
 22 gennaio 2014

Esercizio -1: Essere iscritti su AlmaEsami per svolgere questa prova.

Esercizio 0: Scrivere correttamente nome, cognome, matricola e posizione in tutti i fogli prima di svolgere ogni altro esercizio. Scrivere esclusivamente a penna senza abrasioni. E' vietato l'uso delle penne cancellabili, della matita, dei coprenti bianchi per la correzione (bianchetto) e la scrittura in colore rosso (riservato alla correzione).
 Il compito e' formato da tre fogli, sei facciate compresa questa. Le soluzioni che si vogliono sottoporre per la correzione devono essere scritte negli spazi bianchi di questi fogli. Non verranno corretti altri supporti.
 E' obbligatorio consegnare il compito, e' possibile chiedere che esso non venga valutato scrivendo "NON VALUTARE" in modo ben visibile nella prima facciata.
 Per svolgere questo compito occorre solo una penna e un documento di identità valido. La consultazione o anche solo la disponibilità di altro materiale comporterà l'annullamento del compito (verrà automaticamente valutato gravemente insufficiente).

Esercizio c.1: Per colpa delle frane un ponte veicolare e' rimasto danneggiato. I veicoli possono ora percorrerlo solo in senso unico alternato e, per non eccedere la portata, al piu' ci possono essere sul ponte N veicoli contemporaneamente.

Scrivere il monitor bridge sapendo che i veicoli chiameranno il monitor prima di entrare nel ponte e dopo averlo attraversato. I veicoli che entrano dal lato est escono dal lato ovest e viceversa. Quindi i (numerati) veicoli eseguiranno uno dei due frammenti di codice seguenti:

<pre>bridge.enter(E) /* cross the bridge */ bridge.exit(W)</pre>	<pre>bridge.enter(W) /* cross the bridge */ bridge.exit(E)</pre>
--	--

La soluzione proposta deve evitare deadlock e deve essere efficiente. (controesempio: una soluzione che prevedesse l'attraversamento del ponte da parte di un solo veicolo alla volta sebbene rispetti tutti i vincoli sarebbe considerata errata poiche' inefficiente).

Oltre al codice si richiede una descrizione degli accorgimenti adottati per rendere la soluzione efficiente.

Esercizio c.2:

Sia dato un servizio di message passing asincrono che preveda le seguenti chiamate:

```
void asendx(pid_t destination, message msg)
message arecvx(void)
```

Rispetto al classico servizio di message passing asincrono la funzione arecvx non prevede la selezione del mittente, restituisce il primo messaggio ricevuto.

a. Questo esercizio chiede di implementare su di esso la funzione

```
message msgxchange(pid_t pid, message msg)
```

che realizzi un servizio di scambio sincrono di messaggi. Un processo p che chiami la msgxchange specificando come parametri l'identificativo di un processo q e un messaggio, deve sospendersi fino a quando il processo q non abbia similmente chiamato la msgexchange indicando l'identificativo di p come parametro. Il messaggio passato come parametro da p deve essere il valore di ritorno della funzione per q e, viceversa, il messaggio passato come parametro da q deve essere il valore di ritorno per p.

b. Date le specifiche del punto a., e' possibile che si generino casi di deadlock?

Esercizio g.1: Sia dato l'algoritmo di rimpiazzamento MAXNUM. Come pagina da rimpiazzare MAXNUM sceglie sempre quella con l'indirizzo logico piu' alto (numero di pagina maggiore).

a. mostrare una stringa di riferimenti di lunghezza infinita e che generi infiniti page fault tale che MIN e MAXNUM si comportino esattamente nello stesso modo

b. mostrare una stringa di riferimenti di lunghezza infinita tale che MAXNUM generi un page fault ad ogni accesso in memoria mentre MIN generi un numero finito di page fault

(in entrambi i punti l'insieme delle pagine utilizzate nelle stringhe di riferimenti deve essere finito)

Esercizio g.2: Rispondere alle seguenti domande:

1. La paginazione non elimina completamente ne' la frammentazione interna ne' quella esterna ma le rende trascurabili. Perche'?
2. Il file system ext2 e' piu' efficiente nell'accesso diretto a file di piccole dimensioni rispetto a quelli di grandi dimensioni. Perche'?
3. In quali casi l'algoritmo C-Look e' preferibile all'algoritmo Look? Perche'?
4. Nella memorizzazione delle capability per il controllo di accesso ai file puo' essere usato un algoritmo di crittografia a singola chiave (chiave privata). Perche'?